NPS-CS-02-006

NAVAL POSTGRADUATE SCHOOL
Monterey, California

MYSEA Security Architecture

by

Cynthia E. Irvine
David J. Shifflett
Paul C. Clark
Timothy E. Levin
George W. Dinolt

May 2002

Approved for public release; distribution is unlimited.

Prepared for: Center for INFOSEC Studies and Research

NAVAL POSTGRADUATE SCHOOL
Monterey, California 93943-5000

RADM Admiral David R. Ellison
Superintendent

R. Elster
Provost

This report was prepared for and funded by the Naval Postgraduate School Center for Information Systems Security (INFOSEC) Studies and Research (NPS CISR) .

This report was prepared by:

Cynthia E. Irvine
Associate Professor

David J. Shifflett
Research Associate

Paul C. Clark
Research Associate Professor

Timothy E. Levin
Research Associate Professor

George W. Dinolt
Associate Professor

Reviewed by:

Neil C. Rowe
Professor
Department of Computer Science

Christopher Eagle, Chair
Department of Computer Science

Released by:

D. W. Netzer
Associate Provost and
Dean of Research

REPORT DOCUMENTATION PAGE

Form approved

OMB No 0704-0188

1. AGENCY USE ONLY (Leave blank)

2. REPORT DATE
May 2002

3. REPORT TYPE AND DATES COVERED
Technical Report

4. TITLE AND SUBTITLE

MYSEA Security Architecture

5. FUNDING

MIPR No. 00-E583

6. AUTHOR(S)

Cynthia E. Irvine, David J. Shifflett, Paul C. Clark, Timothy E. Levin, George W. Dinolt

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Center for Information Systems Security Studies and Research (CISR)
Naval Postgraduate School, 833 Dyer Road, Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

NPS-CS-02-006

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
DARPA/ATO
CHATS Program
3701 North Fairfax Drive
Arlington, VA 22203-1714

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words.)

We describe an innovative architecture consisting of trusted security services and integrated operating system mechanisms for the protection of distributed multi-domain computing environments from malicious code and other attacks. These security services and mechanisms extend and interoperate with existing workstations, applications and open source operating systems, providing new capabilities for composing secure distributed systems using commercial off-the-shelf (COTS) components. The latter construct results from the realization that unless a secure system offers users comfortable and familiar interfaces for handling routine information, the secure system will fail due to lack of user acceptability.

14. SUBJECT TERMS

Information assurance, computer security, trusted computing, high assurance, operating system, open source

15. NUMBER OF PAGES
20

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Unclassified

NSN 7540-01-280-5800

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std 239-18

MYSEA Security Architecture[i]

Cynthia E. Irvine, David J. Shifflett, Paul C. Clark, Timothy E. Levin, George W. Dinolt

Center for Information Systems Security Studies and Research
Computer Science Department
Naval Postgraduate School
Monterey, California 93943

Abstract

We describe an innovative architecture consisting of trusted security services and integrated operating system mechanisms for the protection of distributed multi-domain computing environments from malicious code and other attacks. These security services and mechanisms extend and interoperate with existing workstations, applications and open source operating systems, providing new capabilities for composing secure distributed systems using commercial off-the-shelf (COTS) components. The latter construct results from the realization that unless a secure system offers users comfortable and familiar interfaces for handling routine information, the secure system will fail due to lack of user acceptability.

1. Introduction

The U.S. Department of Defense (DoD) computer systems and networks are highly dependent on the security and functionality of a National Information Infrastructure that, as currently organized, does not provide adequate defense against constant and increasingly sophisticated attacks. As a consequence, we risk corruption of critical data and

[i] The views expressed in this paper are those of the authors and should not be construed to reflect those of their employers or the Department of Defense. This work was supported in part by the MYSEA project of the DARPA/ATO CHATS program.

systems, leakage of sensitive information, and degradation of service to fundamental defense systems. Industrial systems run the risk of economic espionage, while the lack of policy-enabled Joint Command and Control Systems constrains military operations. The types of attacks that can be mounted against modern systems range from trivial to serious. A synopsis of attacks is illustrated in Table 1.

Table 1. Attack Elements and System Assurance Required for Defense

Attack Motive	Attack Strategy	Attack Resources	Threat	Assurance Required
Political Military	Long Term Planning	Well Funded	System Subversion	Highest
Political Military	Mid Term Planning	Modest to High Funding	Malicious Code/Trojan Horses	High
Malicious Amusement	Short Term Planning	Low to Modest	Flaw Exploitation	Modest
Malicious Amusement	Ad Hoc	Low	Interface Exploitation	Low

To secure mission critical information systems for the DoD and the nation, new trusted computing approaches are required, involving both interoperable system security features and standardized security mechanisms. We describe an innovative architecture to provide trusted security services and integrated operating system mechanisms that can protect distributed multi-domain computing environments from malicious code and other attacks. These security services and mechanisms extend and interoperate with existing applications and open source operating systems, providing new capabilities for composing secure distributed systems using commercial off-the-shelf (COTS) components. The latter objective results from the realization that unless a secure system offers users comfortable and familiar interfaces they use when handling routine information, the secure system will fail due to lack of user acceptability.

The purpose of the Monterey Security Enhanced Architecture (*MYSEA*, pronounced, `my-SEE-ah`) architecture is to provide a trusted distributed operating environment for enforcing multi-domain security policies, which supports unmodified COTS productivity applications. The architecture encompasses a combination of many low-assurance

commercial components and relatively few specialized (e.g., high-assurance) multi-domain components. This arrangement permits the ongoing DoD and U.S. Government investment in commodity personal computer (PC) operating systems and applications to be integrated into an environment where enforcement of critical security policies is assigned to more trusted elements. Assurance is derived from the application of high assurance system design and development methods to the trusted elements as well as to the overall architecture.

The trusted computing base (TCB) for MYSEA is called the Monterey Secure Architecture Operating System (*MYSEOS*, pronounced, `my-SEE-ose`). In the MYSEA prototype we have constructed, MYSEOS is based upon a security-enhanced version of the OpenBSD operating system. However, the operating system modifications we have defined are modular and conceptually simple enough that they could be accomplished on a variety of open source platforms (e.g., Linux), while the architecture can support higher assurance TCB components, as well[ii]. We also provide a mechanism for vertical integration of application security requirements with underlying security services, applying an existing Quality of Security Service model and framework [24] to the integrated security structure. Additionally, the MYSEA system supports secure *trusted path*[iii] communications between the user and the trusted OS.

Several aspects of this research provide innovative advances in the state of the art for protecting multiple domains of information and for the management of security policies and security services in support of critical applications. Ultimately, the commercial proliferation of these innovations will be available for direct consumption by the DoD for use by operational forces as well as for critical national information infrastructure systems. Specific innovations that we anticipate to be suitable for immediate technical transfer to commercial products are:

- A distributed architecture for isolating trusted components in support of commercial and open source applications. The innovative use of add-on components in commercial client-server systems can potentially magnify the impact of trusted open source systems.
- An open source trusted path mechanism for assured and unambiguous user communication with the trusted

[ii] The use of an unevaluated (and possibly unevaluatable) operating system as the TCB cannot achieve the assurance required for the secure management of information having a range of sensitivity levels [33].

[iii] A trusted path provides an unforgeable bidirectional connection between the user and trusted elements of the system.

computing base.

- Techniques for vertical integration of security policy control functions with underlying security services in a Quality of Security Service framework.
- Single sign-on for access to a community of distributed multi-domain policy servers. Once a user has authenticated to MYSEOS, application sessions may be transferred to any confederated MYSEA Server.

2. Monterey Security Enhanced Architecture

MYSEA is a distributed client-server architecture featuring a combination of (relatively few) specialized policy enforcing components and multiple open source and commercial off-the-shelf components. The major physical components of the architecture are illustrated in Figure 1:

- Security enhanced servers which provide the locus for security policy enforcement and host various open source or commercial application protocol servers, and
- Security enhanced workstations that consist of commercial-class PCs executing popular commercial software products, along with Trusted Path Extensions that provide trustworthy policy support mechanisms and thus permit server-enforced security policy to be distributed across the network.

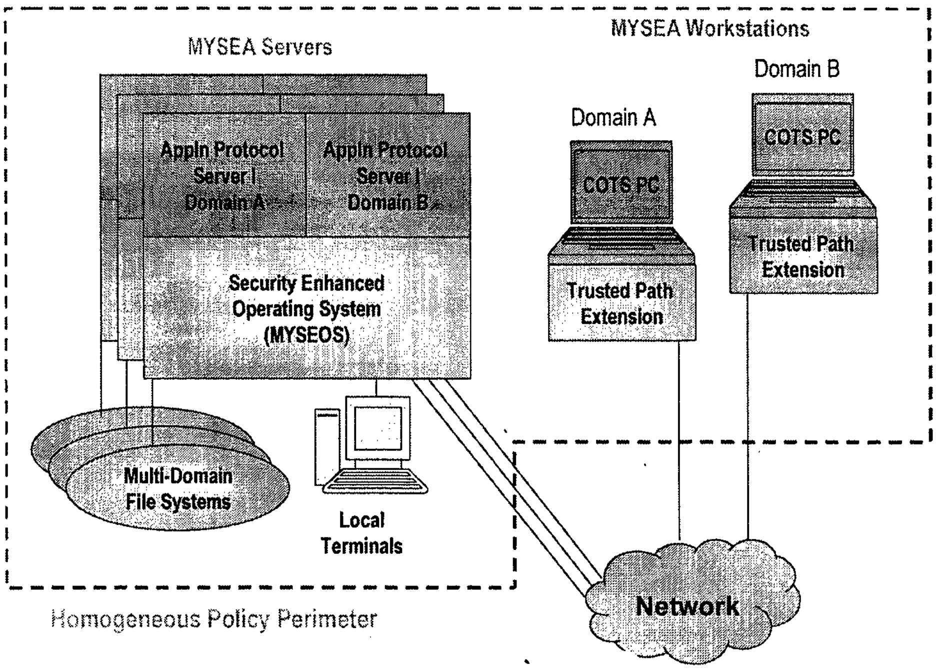

Figure 1. Monterey Security Enhanced Architecture (MYSEA)

The MYSEA Server enforces the security policy and controls access to information. At its heart is a security-enhanced version of the OpenBSD operating system (MYSEOS). Application protocol servers run on the trusted server and provide services and interfaces to shared resources. When MYSEOS is combined with untrusted, but policy constrained (and, in some instances, policy aware) application protocol servers, the result is the MYSEA Server. Each MYSEA workstation is a PC equipped with a Trusted Path Extension device that provides MYSEA policy support at the workstation. The MYSEA Server(s) and the Trusted Path Extension(s) are the only components directly connected to the physical network. Multiple MYSEA Servers provide scalability within the desired security policy perimeter.

MYSEA Concept of Operation

Using the Trusted Path Extension at the PC, users log on to the MYSEA system by way of a trusted path, establishing an identity for audit and access control purposes, and then establish session properties such as current sensitivity level. Subsequently, the user can log on to the native client OS at the PC and use standard commercial client software (e.g., web browser or e-mail program) to access applications supported by the MYSEA Server, or use

any applications supported by the local PC. From the PC the user can access any domain of server data allowed by the security policy (for example, reading domains of data that are lower in sensitivity that the negotiated session level) as well as access local data. By again invoking the trusted path, the user can request to modify session security attributes, such as "session level." During such negotiations, the Trusted Path Extension will ensure that client access to the network is blocked.

MYSEA Components

The MYSEA system consists of the following hierarchy of components, which are described below.

- MYSEA Server
    - Policy-aware application protocol servers
    - MYSEOS
        - trusted path services
        - Security Support Services
        - secure session services
        - quality of security services
        - cryptographic services
        - multi-domain open source kernel (MLS-enhanced OpenBSD)
- MYSEA Workstation
    - Trusted Path Extension
    - COTS PC, including unmodified:
        - operating system
        - user interface
        - application's
        - network connections

MYSEA Server

Each MYSEA Server consists of MYSEOS, which enforces critical security policy, and assorted untrusted application server instances (e.g. one per security domain per user). The actions of the application servers are constrained by the policy enforcement mechanisms of MYSEOS. The application servers are functionally equivalent in terms of overall application-level protocol support to a COTS application server for the particular protocol provided. Thus, each application server is compatible with existing COTS client packages. Additionally,

information managed by application servers can be organized to support such sharing as is allowed by the server, as well as advisory labeling.

MYSEOS

MYSEOS (depicted in Figure 2) is built on OpenBSD as a set of kernel enhancements to create labeled protection domains and a set of additional security services. The MYSEOS kernel associates security attributes with active and passive entities exported at the operating system interface. Enhancements include a protected security manager configured to interpret these attributes and enforce policy according to configuration-specific rules. An important policy for the MYSEOS kernel to enforce is that malicious code may neither exfiltrate confidentially-sensitive data nor corrupt information of higher integrity; to support this, the MYSEOS kernel provides multi-domain file system support, which provides for the global and persistent separation of data into its respective domains. Other security services that have been integrated into the MYSEOS kernel are described below.

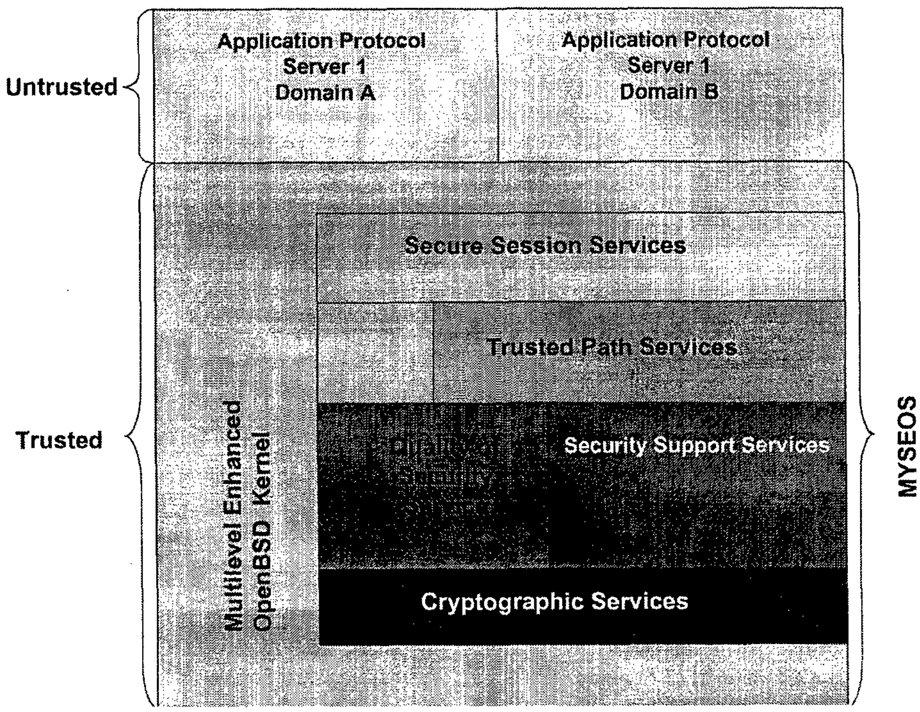

Figure 2. MYSEA Server

Trusted Path Services

The Trusted Path Services component supports multiple locally attached terminals, as well as multiple remote MYSEA workstations. Trusted Path Services maintains the state of the user-to-MYSEA interaction, for example, a user may be logged in with default security attributes, but may not have started a session executing untrusted application code. Trusted Path Services provides an interface to the Security Support Services component to support identification and authentication, negotiation of domain or domain range, password modification, account creation and deletion, and user security attribute maintenance. Once a session has been established, the Trusted Path Services provides a distributed Session Status Database to the Secure Session Services component.

Secure Session Services

The Secure Session Services component is used to launch instances of untrusted, constrained application protocol servers. It provides trusted policy-sensitive services, with functionality similar to that of classic *inetd* implementations and supports standard application protocol transmissions. The Secure Session Services accesses the Session Status Database, maintained by the Trusted Path component, to determine the security attributes to associate with each application protocol server.

This Session Status Database contains tuples that uniquely identify the user, the client workstation associated with the user, the status of the user session, the security attributes of the session, and other security relevant information. Through a session status communication mechanism, information in the Session Status Database can be provided to distributed multi-policy platforms, thus providing a single sign-on and session level capability.

Quality of Security Service Support

MYSEA can be integrated with an external resource or QoS manager to provide a means of dynamically managing its security and performance characteristics. The MYSEA QoSS Manager is the external QoSS interface to MYSEA, and governs security and performance factors of the various MYSEA components, for example, which application protocol servers the client may interact with, and the cryptographic protection characteristics of the underlying communication channels. The QoSS security and connectivity database is managed by the QoSS manager on the MYSEA server, and is distributed to the Trusted Path Extensions, as needed.

The Quality of Security Service manager provides a user interface so that decision makers can request the overall security posture of the network. This interface provides the decision maker with a simple set of choices, hiding the underlying complexity of the quality of security service mechanisms [32].

Constrained Application Protocol Servers

The secure session server provides instances of standard protocol servers for each client or for equivalence classes of clients. The Session Status Database, which is managed by the trusted path services component, but is readable by the secure session server, is used to assign security attributes to protocol servers launched on behalf of a requesting client. Thus the protocol servers are associated with domains reflecting the granularity of the policy enforced by the underlying trusted operating system.

Protocol servers take two forms. The first form is a standard, policy-unaware protocol server, e.g. HTTP. These servers are restricted to accessing files and other objects associated only with the particular domain associated with the session. The second type of server is policy-aware, e,g, a file system, [22] and is able to take advantage of certain security policy domain relations that permit limited modes of access to certain other domains (e.g., "read down" for mandatory confidentiality policies).

Among the application servers we have adapted to the MYSEA environment are: Internet Mail Access Protocol (IMAP) based on the University of Washington IMAP server [15], Hypertext Transfer Protocol (HTTP) based on the Apache server [6], and Simple Mail Transfer Protocol (SMTP) based upon sendmail [9]. Each server required little or no code modification to be adapted to the multilevel environment. With a proper configuration of the policy-aware application protocol server, users can view information at or below their current session levels.

MYSEA Workstations

Platforms may be considered to be an automated extension of the individual using them. In a network, a user may have a client workstation that is being used to access a server across the network. From the simplest point of view, the client system can be viewed as containing the following elements: processing services, user interface services, and I/O services supporting network communications and storage.

MYSEA workstations consist of two physical components: a Trusted Path Extension and an untrusted personal computer (see Figure 3). The PCs are typical COTS products hosting a popular commercial operating system and a commercial application suite. The application suite contains client software intended to access standard application protocol servers. For example, mail service clients might include: Lotus Notes, Outlook, Pine, Postal, and Netscape[17]. A typical browser supports the client interface to web pages.

To ensure that object reuse requirements are met, workstations are managed to be, in effect, "diskless," with sufficient volatile RAM-disk capability to support a wide variety of user applications. The Trusted Path Extension satisfies object reuse requirements by ensuring that RÀM and other volatile primary and secondary storage are purged with each change of session level or new user login at the workstation.

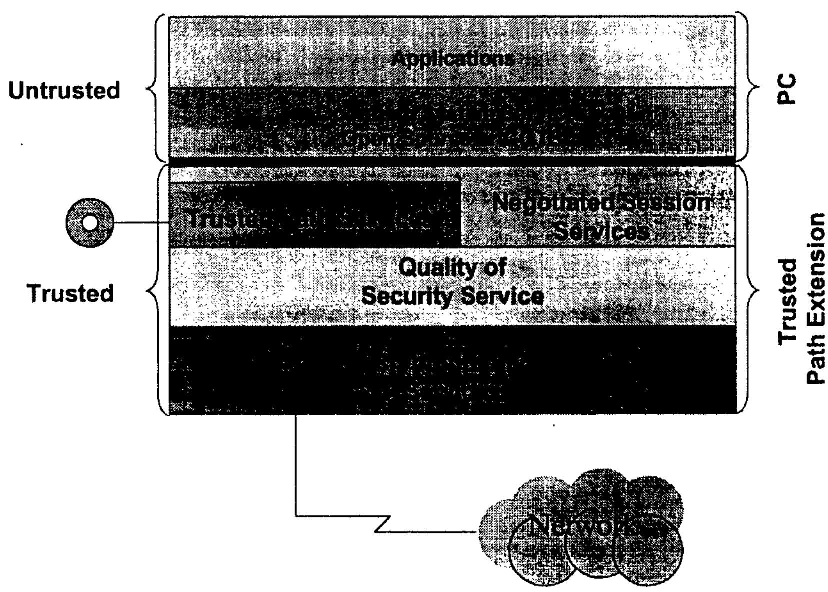

Figure 3. MYSEA Workstation

Trusted Path Extension

The trusted elements of the MYSEA system provide the locus of security policy enforcement. Not only do these elements provide runtime policy enforcement, but they must also provide services for the enforcement of supporting policies. To create a distributed TCB, the architecture includes a Trusted Path Extension at each workstation.

The Trusted Path Extension maintains its own self-protecting domain that is separate from the user and workstation domains. The use of a separate processor for the Trusted Path Extension ensures that it cannot be subverted by malicious software on the workstation. Architecturally, the Trusted Path Extension provides the PC's only access to the network.

The Trusted Path Extension has two form factors: an internal PCI card (planned for future development) and an external hand-held computer (per the current MYSEA prototype). In the PCI card format the Trusted Path Extension presents a NIC interface to the workstation. User trusted path I/O, including the secure attention key, is achieved via strictly controlled access to the PC keyboard and display. In the handheld format, the Trusted Path Extension performs IP network address translation for all IP traffic going between the PC and the LAN -- and user trusted path I/O occurs via the handheld's native keyboard and screen.

Simplicity has been a primary design goal for the Trusted Path Extension. The objective was not to construct a second operating system for the PC; it does not require the complexity and rich set of services provided by a typical PC (e.g. file system, printers and other peripheral drivers). The Trusted Path Extension can be viewed as a minimized embedded system that maintains no state of its own; instead, it functions as a "drone" in response to commands from the MYSEA server for controlling the workstation and managing I/O with the user. The Trusted Path Extension, under direction from the MYSEA server, supports the following services:

- Secure Attention Key – this service permits users to initiate unambiguous communication with MYSEOS for unspoofable presentation and capture of security critical data at the user interface. The secure attention key must cause a state change in the Trusted Path Extension such that an unforgeable communications path (viz. a *trusted path*) to MYSEOS is established.
- Trusted Path Services – when the trusted path is invoked, the user may elect to input security critical information, such as a password. The trusted path services ensure that prompts from the server are displayed and that an input mechanism for replies is available.
- Controlled LAN Access – provide non-bypassable, controlled access to the LAN from the PC. Malicious

software on the PC cannot bypass the Trusted Path.

- Communications and cryptographic services – provide protected communication channels between the server and the Trusted Path Extension. These protected communications are based upon protocols that support both the establishment and maintenance of a trusted path and session-level communications, such as to initiate communication with the server (via the secure attention key), as well as to receive and to respond to commands from the MYSEA Server.

- Negotiated Session Services – these mechanisms ensure trusted *object reuse* at the client PC for both primary and secondary storage. When a user chooses to change domains, certain policies require that information associated with the previous domain be purged from the untrusted PC, e.g. previous session information cannot be reused by subsequent sessions in conflict with the distributed security policy. The Trusted Path Extension ensures that object reuse requirements are met with each session change and as dictated by policy for session level changes. The Trusted Path Extension supports object reuse directives issued by MYSEOS. These directives may include both functional and procedural actions at the workstation.

- Control of Security Critical Activities –control the client and its resources at the time of boot and control security critical actions throughout the client session.

- Quality of Security Service - as networks become more complex and adaptive, it may be necessary to provide "security on demand." When conditions on the network change, requirements for security may also change. In response to a change notification, quality of security service mechanisms located on the Trusted Path Extension can modify the protection services afforded an ongoing session. The selection of protection mechanisms for communications between the client and the server may be based upon network conditions such as INFOCON mode. A version of IPSec adapted to provide automated, dynamic Quality of Security Service through the use of an enhanced version of a policy server such as Keynote [7]permits selection of protection mechanisms for MYSEA Servers.

3. MYSEA Developmental Assurance

Our rigorous security engineering and development process [25] is intended to support high assurance evaluation of (portions of) the finished product. Development begins with the capture of the security policy to be enforced and an interpretation of that policy in terms of an abstract computer system. This may produce a formal security policy

model and subsequent evidence that policy enforcement objectives are met. In parallel with that formal approach, the engineering team develops a series of specifications that ranges from threat model and high level requirements to detailed implementation documents and code. A system requirements specification for a secure system incorporates security considerations in conjunction with all other requirements.

Starting with a threat model and a system requirements specification, we develop a system architecture. From these, we develop functional specifications for specific components, such as the protection module of the distributed trusted OS, and a corresponding detailed design specification for those same components. Concurrent development of requirements, functional, and design specification allow us to identify notions that are generalizable and can be abstracted for inclusion in the higher-level documents. Conversely, detailed items more appropriate for the lower-level specification can be moved down. This iterative feedback approach permits us to develop documents suitable for evolutionary engineering processes [5][31] as experiential or environmental factors lead to requirements for new versions of the system.

4. Related Work

The research defined in this paper builds on a variety of previous efforts. The primary work we are extending is from the MLS LAN project [2][4][10][11][12][15][17][20][23][36][44]. This previous project resulted in development of networking modules to support the following functions: (1) a trusted path between client workstations and the server, (2) session-level negotiation at the server from the client workstations, and (3) secure single-level session communications on the Ethernet for client workstations at different session levels (i.e., different domains communicate with the server through a single physical network device). In the MYSEA project, we have adapted and extended the fundamental research underlying these concepts and provided prototype demonstrations of the integrated concepts in an open-source environment.

User access to multi-domain data via commercial workstations and applications

Hinke suggested the notion of a high assurance server to provide a locus of multi-domain control to single level clients [21]. In that design sketch, clients were relegated to a single level and were connected to the multilevel server via single level network links. Although this architecture may be useful in certain static situations, it does not

provide the flexibility inherent in the MYSEA design. By restricting the client to a single level throughout its lifetime, users are required to access multiple clients in order to manipulate information at several levels. In contrast to this approach, the MYSEA architecture allows clients to renegotiate session levels.

Rushby and Randell [38] describe a design for a distributed secure system that utilizes "trusted network interface units" (TNIUs) to connect workstations at different access classes to a local area network, through which access to a distributed multilevel file server is provided. Identification and authentication of users, as well as session level negotiation via the TNIUs is also described. Over and above this functionality, the MYSEA architecture also allows a more general purpose client-server operating environment, whereby new application servers can be easily added to the system, and *thin clients* may also be easily supported.

Replication architectures [18] provide a simple technique to achieve near-term multilevel security by copying all information at low security levels to all dominating levels. On a small scale, one can expect them to work rather well; on a large scale, in terms of both numbers of documents to be replicated and numbers of security levels to be replicated to, their usefulness is rather problematic. The preponderance of information used in the DoD today is either unclassified or designated sensitive but unclassified (SBU). Replication of this large amount of data to all higher levels seems infeasible. In the commercial sector, the ratio of proprietary to less sensitive information is similar. The MYSEA multi-domain solution does not utilize replication as a fundamental mechanism, so it avoids these problems.

Various *virtual machine monitor* approaches have been suggested [8][28][3] for supporting COTS applications while reliably separating different domains of data. In general, for these approaches to be trustworthy requires both the use of hardware that is strictly virtualizable [19], and a trustworthy monitor mechanism for separating the activities of the virtual machines. Creating a monitor that is trusted enough to both separate different domains of activity, and allow read-down to less sensitive domains (as does MYSEA) is all the more difficult. While [28] was designed to provide high assurance read-down capabilities, the effort was cancelled for lack of commercial support. The VMM approach continues to be problematic for separation of different domains of data because many current microprocessors are not strictly virtualizable [35], leading to complex software solutions, and because of the difficulty of creating a trusted monitor.

Non-distributed approaches to supporting access to multi-domain data via COTS applications have been proposed in the Seaview project [13][30], the "Purple Pennelope" project [34], and some VMM architectures (see above). In each of these approaches, a separate process is created for each security level. Purple Pennelope has limited assurance, as it runs as a user-level application, and it does not support a modifiable session level. The others are supported by an underlying reference validation mechanism that controls access to multi-domain data. The MYSEA project extends certain concepts from these projects into a distributed environment.

The Naval Research Laboratory (NRL) Network Pump [26] was developed to allow messages from a system operating in a low security domain to be sent to a system operating in a high security domain, and to prohibit messages and other information from going in the reverse direction. Additionally, the NRL Pump has been proposed as part of an overall network architecture to provide a more general two-way connectivity between multiple subnets at different security levels, resulting in a multiple single-level (MSL) network [27]. In this approach, information is also processed by an automated filter-guard to allow policy-approved information to flow from higher domains to lower domains. The MSL network approach has several drawbacks that the MYSEA avoids:

- The capital and administrative cost of separately maintained local area networks (LANs)
- The technical challenge of providing an automatic and reliable information filtering mechanism
- The cost of maintaining filtering rules for changing policies (e.g., the policy may evolve as administrators become aware of different threats)
- The technical challenge of filtering complex information structures, such as multimedia.

The Starlight project [1] was designed to support logically separate single-level workstations connected by a switch to data management subsystems at different (single) levels. Software associated with the switch ensures that the current level of the workstation matches the level of data subsystem indicated by the switch setting. Starlight also allows low confidentiality information to flow through the switch to high sessions, providing a "read-down" capability. This approach has the same basic drawbacks as the MSL network, described above.

Novell Trusted Workstation Partnership [16] defined a network architecture for separating clients in different security domains with their Class C2 evaluated network software. An instantiation of this approach utilized the

Sistex, Inc., Assure EC plug-in card to separate the different file system domains, however, this product is no longer marketed, and detailed documentation is not available.

Other Open Source Multi-Domain Variations

The "rule set based access control'" (RSBAC) system [37] is a Linux extension wherein all security relevant system calls are routed through a central decision component. Access-control decisions are based on the type of access and on attributes attached to the calling subject and to the target to be accessed. The robustness and security characteristics of this implementation are not clear from the documentation [11].

The Safe Areas of Computation project from the University of Santa Barbara [14] defines a distributed architecture to support secure access to multiple data domains. In this approach, a trusted component on each client and server platform is responsible for the access control decisions of that platform. The trusted component is assisted by an untrusted/unprotected communication package that manages metadata exchange with the corresponding (reciprocal) client or server, and participates in encryption, decryption and key management activities. For client platforms, it is envisioned that the access control component can be implemented in a smart card. This approach requires modification to client applications to interact with the communications package, so it does not meet requirements for use with MYSEA systems. The communications package may also be a vulnerable point of attack, with regard to cryptographic processing.

The Security-Enhanced Linux project from the NSA has recently released some information regarding their approach to controlling multiple information domains in an open source operating system [29] [40]. From a preliminary examination of these papers, it appears that the Security-Enhanced Linux project has not yet defined several mechanisms that are planned for MYSEA:

- Remote-client login to the trusted OS
- Trusted path communications with the trusted OS
- Changing a user session security level
- A mechanism for assigning security-domain context to a newly received network connection
- Trusted, rather than client, support for IPsec message labeling.

- Support for untrusted clients, i.e., clients who are not based on Security-Enhanced Linux.

Trusted Path

"Trusted path" refers to mechanisms that provide assurance that security-critical functions are provided by the "real" system rather than masquerading software. Several commercial systems have implemented trusted path mechanisms, including Windows NT [42], Trusted Solaris 7[41], and the XTS-300 [43]. In the case of NT and Solaris, it is notable that the processing of security requests is handled, at least partially, outside of the kernel, so the assurance of request handling is of some question.

5. Conclusion

We have presented the Monterey Security Enhanced Architecture (MYSEA), which provides a trusted distributed operating environment for enforcing multi-domain security policies, and which supports unmodified COTS productivity applications. The architecture encompasses a combination of many (untrusted) commercial components and relatively few trusted multi-domain components. Our prototype implementation utilizes a security-enhanced version of the OpenBSD operating system, called *MYSEOS,* as the policy enforcing *trusted computing base* (TCB). The architecture is general enough that it would easily accommodate a high assurance TCB, as well.

MYSEA introduces several innovations for protecting multiple data domains and for managing security policies and security services in support of critical applications, including:

- A distributed trusted architecture that utilizes commercial and open source applications to access multiple data domains.
- An open source trusted path mechanism.
- Techniques for vertical integration of security policy control functions with underlying security services.
- Single sign-on for access to a community of distributed multi-domain policy servers.

In the future, we plan several additions and enhancements to MYSEA. We have begun investigation of a ring mechanism[39] for open source operating systems, to help constrain the behavior of applications that run on MYSEOS and similar environments.

There are various systems and tools available to support the automated verification of computer system behavior. As a precursor to the formal analysis of the security behavior of MYSEA components, we have received support to perform a survey of available formal verification tools. That survey was started this summer.

We have recently started a project that includes the development of a very high assurance micro kernel. The goal for the Trusted Computing Exemplar Project is to provide a worked example of a high assurance system that can be used by the education community, government and industry. To further that aim, we plan to make the micro kernel, its development methodology and its evaluation evidence generally available through open source methods. As an early example of the application of the high assurance micro kernel, we plan to implement a high assurance Trusted Path Extension.

References

[1] Anderson, M., North, C., Griffin, J., Milner, R., Yesbert, J., and Yiu, K., "Starlight"Interactive Link, Proceedings of the 12th Computer Security Applications Conference, San Diego, CA December 1996.

[2] Balmer, Steven, Trusted Computing Base Extension Control System For Client Workstations, Masters Thesis, Naval Postgraduate School, Monterey, California, September 1999.

[3] Balmer, S. R., and Irvine, C. E. "Analysis of Terminal Server Architectures for Thin Clients in a High Assurance Network," Proceedings of the National Information Systems Security Conference, Baltimore, MD, October 2000, pp 192-202. (Best Paper Award).

[4] Bartram, Scott, Supporting a Trusted Path for the Linux Operating Sytem, Masters Thesis, Naval Postgraduate School, Monterey, California, June 2000. (co-advised with Paul C. Clark)

[5] Berzins, V., and Luqi, Software Engineering with Abstractions, Addison-Wesley, Reading, MA, 1990.

[6] Bersack, Evelyn, Implementation of a HTTP (Web) Server on a High Assurance Multilevel Secure Platform, Masters Thesis, Naval Postgraduate School, Monterey, California, December 2000.

[7] Blaze, Matt, Feigenbaum, Joan, and Keromytis, Angelos D., KeyNote: Trust Management for Public-Key Infrastructures, In Proceedings of the 1998 Security Protocols International Workshop, Springer LNCS vol. 1550, pp. 59 - 63. April 1998, Cambridge, England. Also AT&T Technical Report 98.11.1.

[8] T. Borden, J. Hennessy and J. Rymarczyk. Multiple Operating Systems On One Processor Complex, IBM Systems Journal, 28(1):104-123, 1989

[9] Brown, Emma, SMPT on a High Assurance Mulitlevel Server, Masters Thesis, Naval Postgraduate School, Monterey, California, September 2000.

[10] Bryer-Joyner, Susan, and Heller, Scott D., A Protocol for Establishing a Trusted Path Over an Untrusted Local Area Network, Masters Thesis, Naval Postgraduate School, Monterey, California, March 1999. Navy League Award received by LT Heller upon graduation.

[11] Clark, Paul, Supporting Mandatory Access Control in an Educational Environment, Masters Thesis, Naval Postgraduate School, Monterey, California, September 1999.

[12] Clark, P., "Supporting Mandatory Access Control in an Educational Environment", Proceedings of the National Information Systems Security Conference, October 2000.

[13] Denning, D., Lunt, T. F., Schell, R. R., Shockley, W. R., and Heckman, M., Security Policy and Interpretation

for a Class A1 Multilevel Secure Relational Database System. In Proceedings 1998 IEEE Symposium on Security and Privacy, Oakland, CA, April 1988.

[14] dos Santos, A. L. M., and Kemmerer, R., Safe Areas of Computation fro Secure Computing with Insecure Applications, Proceedings of the Computer Security Applications Conference, Phoenix, AZ, December 1999, pp 35-44.

[15] Eads, Bradley, Developing a High Assurance Multilevel Mail Server, Masters Thesis, Naval Postgraduate School, Monterey, California, March 1999.

[16] Epstein, J., Grossman, G., and Schell, R.R., Component Architectures for Trusted Netware, Proceedings of the 18th National Information Systems Security, Baltimore, MD, October 1995, pp 455-463.

[17] Everette, Theresa, Enhancement of Internet Message Access Protocol for User-Friendly Multilevel Mail Management, Masters Thesis, Naval Postgraduate School, Monterey, California, September 2000.

[18] Froscher, J., Kang, M., McDermott, J., Costich, O., and Landwehr, C. E., A Practical Approach to High Assurance Multilevel Secure Computing Service, in Proceedings 10th Computer Security Applications Conference, pp 2-11, Orlando, FL, December 1994.

[19] R. Goldberg. Architectural Principles for Virtual Computer Systems. Ph.D. thesis, Harvard University, Cambridge, MA, 1972

[20] Hackerson, Jason, Constructing a Trusted Computing Base Extension in Commercial-Off-the Shelf Personal Computers for Multilevel Secure Local Area Networks, Masters Thesis, Naval Postgraduate School, Monterey, California, September 1998.

[21] Hinke, T., The Trusted Server Approach to Multilevel Security, Proceedings of the Computer Security Applications Conference, December 1990, pp335-341.

[22] Irvine, C. E., Acheson, T.B. and Thompson, M.F., "Building Trust into a Multilevel File System," Proc. of the 13th National Computer Security Conference, October 1990, Washington, DC.

[23] Irvine, C. E., Anderson, J.P., Robb, D.A., and Hackerson, J., "High Assurance Multilevel Services for Off-The-Shelf Workstation Applications," Proceedings of the National Information Systems Security Conference, Crystal City, VA, pp.421-431, October 1998.

[24] Irvine, C. E., and Levin, T., "Quality of Security Service," in the Proceedings of the New Security Paradigms Workshop, September 2000.

[25] Irvine, C. E., Levin, T., Wilson, J. D., Shifflett, D., and Pereira, B., "A Case Study in Security Requirements Engineering for a High Assurance System," in the Proceedings of the Symposium on Requirements Engineering for Information Security, March 2001.

[26] Kang, M., Froscher, J., and Eppinger, B., Toward an Infrastructure for MLS Distributed Computing. In Proceedings of the 14th Annual Computer Security Appplications Conference, pp 91-100, Phoenix, AZ, December 1998.

[27] Kang, M., and Moskowitz, I., Design and Assurance Strategy for the NRL Pump. In IEEE Computer, (31:4), pp 56-64, April 1998.

[28] Karger, P., Zurko, M. E., Bonin, D. W., Mason, A. H., and Kahn, C. E., A VMM Security Kernel for the VAX Architecture. In Proceedings 1990 IEEE Symposium on Research in Security and Privacy, pp 2-19.

[29] Loscocco, P., and Smalley, S., Integrating Flexible Support for Security Policies into the Linux Operating System, http://www.nsa.gov/selinux/slinux-abs.html, October 2000.

[30] Lunt, T. F., Schell, R. R., Shockley, W. R., Heckman, M., and Warren, D., A Near-Term Design for the SeaView Multilevel Database System. In Proceedings IEEE Symposium on Security and Privacy, pp 234-244, Oakland, CA, May 1988.

[31] Luqi, A Graph Model for Software Evolution, IEEE Transactions on Software Engineering, 16(8):917-927, August 1990.

[32] Mohan, Raj, Xml Based Adaptive Ipsec Policy Management In A Trust Management Context, Masters Thesis, Naval Postgraduate School, Monterey, California, September 2002.

[33] National Computer Security Center, Computer Security Requirements Guidance For Applying The Department Of Defense Trusted Computer System Evaluation Criteria In Specific Environments, CSC-STD-004-85, June 1985.

[34] Pomeroy, B. and Weissman, S., Private Desktops and Shared Store, in Proceedings, Computer Security Applications Conference, Phoenix, AZ, December 1998, pp. 190-200.

[35] Robin, J.S. and Irvine, C.E., Analysis of the Intel Pentium's Ability to Support a Secure Virtual Machine Monitor Proceedings of the 9th USENIX Security Symposium, Denver, CO, pp. 129-144, August 2000

[36] Rossetti, Richard K., A Mail File Administration Tool for a Multilevel High Assurance LAN, Masters Thesis, Naval Postgraduate School, Monterey, California, September 2000.

[37] RSBAC, http://agn-www.informatik.uni-hamburg.de/people/lott/rsbac

[38] Rushby, J., Randell, B, A Distributed Secure System. In Proceedings 1983 IEEE Symposium on Security and Privacy, Oakland, CA, April 1983.

[39] Michael D. Schroeder, Jerome H. Saltzer: A Hardware Architecture for Implementing Protection Rings. CACM 15(3): 157-170 (1972).

[40] Smalley, S. and Frazer, T, A Security Policy Configuration for the Security Enhanced Linux, http://www.nsa.gov/selinux/policy-abs.html, October 2000.

[41] Sun Microsystems, Trusted Solaris Security Features User's Guide, Sun Microsystems, Palo Alto, CA, 1994.

[42] TTAP, Windows NT Workstation and Windows NT Server, Version 4.0 with Service Pack 6a and C2 Update, http://www.radium.ncsc.mil/tpep/epl/entries/TTAP-CSC-EPS-99-001.html, November, 1999.

[43] Wang Government Services, Inc, XTS-300 User's Manual, Document ID FS92-393-07, March 1998.

[44] Wilson, J. D., Trusted Networking in a Multilevel Secure Environment, Masters Thesis, Naval Postgraduate School, Monterey, California, June 2000.

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center 8725 John J. Kingman Rd., STE 0944 Ft. Belvoir, VA 22060-6218	2
2.	Dudley Knox Library, Code 013 Naval Postgraduate School Monterey, CA 93943-5100	2
3.	Research Office, Code 09 Naval Postgraduate School Monterey, CA 93943-5138	1
4.	Dr. Douglas Maughan Defense Advanced Research Projects Agency Advanced Technology Office 3701 North Fairfax Drive Arlington, VA 22203-1714	1
5.	Dr. Cynthia E. Irvine Code CS/Ic Department of Computer Science Naval Postgraduate School Monterey, CA 93943-5118	10
6.	Mr. David J. Shifflett Code CS/Sd Department of Computer Science Naval Postgraduate School Monterey, CA 93943-5118	2
7.	Mr. Paul C. Clark Code CS/Cp Department of Computer Science Naval Postgraduate School Monterey, CA 93943-5118	2
8.	Mr. Timothy E. Levin Code CS/Lt Department of Computer Science Naval Postgraduate School Monterey, CA 93943-5118	2

9. Dr. George W. Dinolt 2
Code CS/Dg
Department of Computer Science
Naval Postgraduate School
Monterey, CA 93943-5118